Pertemuan 4

Array pada Java

Objektif :

1. Mahasiswa dapat memahami pengertian Array pada Java
2. Mahasiswa dapat mengetahui bentuk umum dari Array
3. Mahasiswa dapat mengetahui jenis-jenis Array pada Java
4. Mahasiswa dapat mendeklarasikan dan membuat program sederhana menggunakan Array pada Java

P4.1 Teori

1. Pengenalan Array

Pada Pertemuan sebelumnya, kita telah mendiskusikan bagaimana cara pendeklarasian berbagai macam variabel dengan menggunakan tipe data primitif. Dalam pendeklarasian variabel, kita sering menggunakan sebuah tipe data beserta nama variabel atau *identifier* yang unik. Apabila kita ingin menggunakan variabel tersebut, kita akan memanggil dengan nama *identifier*-nya. Sebagai contoh, kita memiliki tiga variabel dengan tipe data int yang memiliki *identifier* berbeda untuk tiap variabel.

```
Int number1;
Int number2;
Int number3;
Number1=1;
Number2=2;
Number3=3;
```

Dari kode di atas, kita tidak perlu mendeklarasikan variable yang banyak dengan tujuan sama. Oleh karena itu digunakan array yang berfungsi mendeklarasikan variable dengan tipe data yang sama. Pada bahas pemrograman Java maupun di bahasa pemrograman yang lain, terdapat sebuah kemampuan untuk menggunakan satu variabel yang dapat menyimpan beberapa data dan memanipulasinya dengan lebih efektif. Tipe variabel inilah yang disebut sebagai **array**. Contoh dari array integer sebagai berikut.

Sebuah array akan menyimpan beberapa item data yang memiliki tipe data sama di dalam sebuah blok memori yang berdekatan yang kemudian dibagai menjadi beberapa ruang. Array adalah sebuah variabel/sebuah lokasi tertentu yang memiliki satu nama sebagai *identifier*, namun identifier ini dapat menyimpan lebih dari sebuah nilai.

2. Pendeklarasian Array

Seperti yang telah dijelaskan secara singkat mengenai *array* pada pertemuansebelumnya adalah kelompok variabel dengan tipe sejenis dan dinyatakan

dengannama yang sama. Dengan kata lain, *array* merupakan tipe khusus yang menyatukan sekelompok variabel dengan tipe yang sama. *Array* di Java dideklarasikan dengan kurung siku: […]. Sintaks umum deklarasi *array*: *type var-name[];*

Atau

type[] var-name;

type berperan dalam mendeklarasikan tipe basis dari *array*. Tipe basis menentukan tipe data bagi masing-masing elemen yang membentuk *array*. Dengan demikian, tipe basis untuk *array* menentukan tipe data yang dimuat oleh *array*.

Berikut contoh-contoh deklarasi *array* di Java:

_ *int numbers[];*

_ *char[] letters;*

_ *long grid[][];*

Pada Java tidak perlu menspesifikasikan ukuran *array* kosong saat mendeklarasikan *array*. Kita harus selalu menyatakan ukuran *array* secara eksplisit saat melakukan **operasi penciptaan** menggunakan operator **new()** atau dengan mendaftarkan item-item untuk *array* pada saat penciptaan.

Berikut contoh pendeklarasian *array* dengan menspesifikasikan ukuran spesifiknya menggunakan operator **new()** dan dengan memberikan daftar item yang termasuk di dalam *array* :

char alphabet[] = new() char [26];

int primes = {7, 11, 13};

3. Jenis-jenis Array pada Java

3.1 Array 1 Dimensi

Array 1 dimensi pada dasarnya senarai (deretan) variabel bertipe serupa. Kita lebih dahulu menciptakan variabel *array* dari tipe yang dikehendaki.

int monthDays[];

Meskipun deklarasi telah menetapkan bahwa **monthDays** adalah variabel *array*, namun sesungguhnya tidak ada *array* yang diciptakan pada saat itu. Nilai dari **monthDays** adalah nilai **null** yang merepresentasikan *array* tanpa nilai dan bukan bernilai nol.

Bentuk umum **new()** untuk *array* 1 dimensi adalah sebagai berikut:

array-var = new() type[size]

Dalam hal ini, *type* menspesifikasikan tipe data yang dialokasikan, *size* menspesifikasikan jumlah dari elemen *array*, dan *array-var* adalah variabel *array* yang dirangkai ke *array*. Dengan demikian, untuk menggunakan **new()**, kita harus menspesifikasikan tipe dan jumlah elemen untuk dialokasikan. Elemen-elemen di *array* yang dialokasikan oleh **new()** akan secara otomatis dinisialisasi ke nol.

Berikut ini salah satu contoh penciptaan untuk *array* yang berisi jumlah hari maksimum di masing-masing bulan:

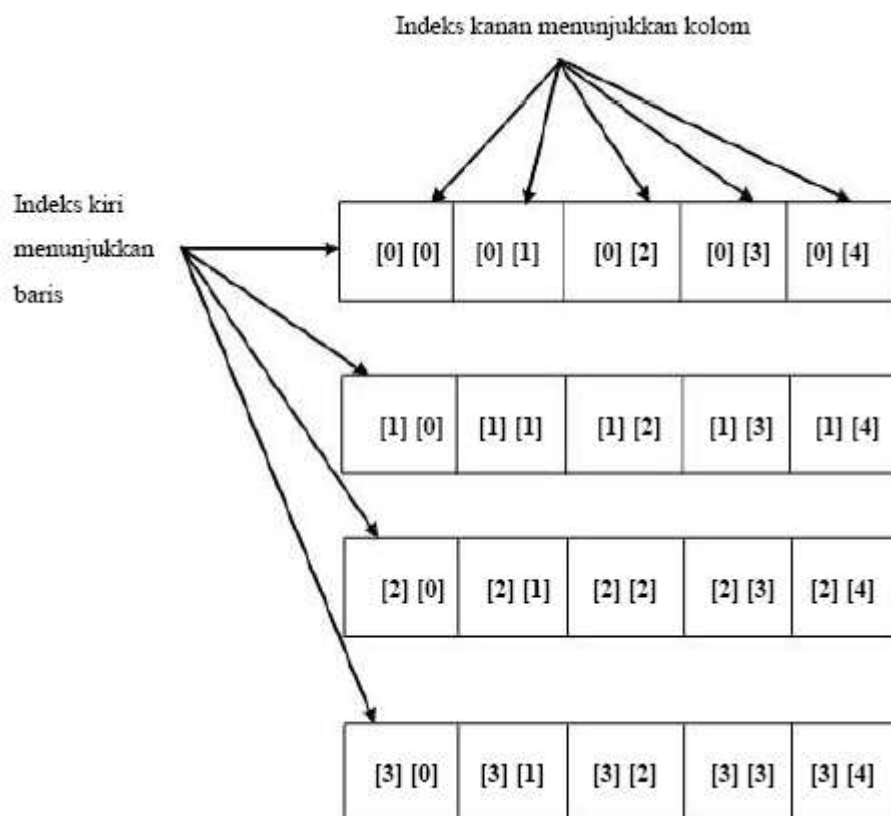
```
class Bulan
{
public static void main(String[] args)
{
// langkah 1: deklarasi variabel array
int monthDays[];
// langkah 2: penciptaan array bilangan int dengan 12 elemen
monthDays = new int[12];
monthDays[0] = 31;
monthDays[1] = 29;
monthDays[2] = 31;
monthDays[3] = 30;
monthDays[4] = 31;
monthDays[5] = 30;
monthDays[6] = 31;
monthDays[7] = 31;
monthDays[8] = 30;
monthDays[9] = 31;
monthDays[10] = 30;
monthDays[11] = 31;
System.out.println("Agustus mempunyai "+monthDays[7]+"
hari");
}
}
```

Hasil (*output*) dari contoh listing program di atas adalah **Agustus mempunyai 31 hari**. Simbol “**//**” (dua garis miring) digunakan untuk suatu komentar (informasi singkat) program, dimana komentar tersebut tidak dieksekusi.

3.2 Array Multidimensi

Di Java, *array* multidimensi sesungguhnya *array* dari *array*. Untuk mendeklarasikan variabel *array*, menspesifikasikan masing-masing indeks menggunakan himpunan kurung siku yang lain. *Array* 2 dimensi bertipe **int** bernama **intArr** dideklarasikan dan diciptakan dengan perintah sebagai berikut:

int int2DArr[][] = new() int [4][5];

Perintah ini mendeklarasikan dan mengalokasikan *array* 4 kali 5 dan memberikan ke variabel **int2DArr**. Secara internal matriks ini diimplementasikan sebagai *array* dari *array* bertipe **int**. Secara konseptual *array* ini digambarkan sebagai berikut:

Array 2 dimensi 4x5

Berikut contoh listing program yang memberikan angka ke masing-masing elemen di *array* dari kiri ke kanan, puncak ke bawah, kemudian menampilkan masing-masing elemen:

```
class Multidimensi
{
public static void main(String[] args){
//Langkah 1: deklarasi variabel array dan penciptaan array
int int2DArr[][] = new int[4][5];
int k=0;
for (int i=0;i<4;i++){
for (int j=0;j<5;j++){
int2DArr[i][j] = k++;
}
}
for (int i=0;i<4;i++){
for (int j=0;j<5;j++){
System.out.print(int2DArr[i][j] + " ");
}
System.out.println();
}
}
}
```

Hasil (*output*) dari contoh listing program di atas:

0 1 2 3 4

5 6 7 8 9

10 11 12 13 14

15 6 17 18 19

4. Pengaksesan Sebuah Elemen Array

Untuk mengakses sebuah elemen dalam array, atau mengakses sebagian dari array, Anda harus menggunakan sebuah angka atau yang disebut sebagai **indeks** atau subscript. Pada saat memasukkan nilai ke dalam array, sebuah **nomor indeks atau subscript** telah diberikan kepada tiap anggota array, sehingga program dan programmer dapat mengakses setiap nilai pada array apabila dibutuhkan. Nilai indeks **selalu dalam tipe integer**, **dimulai dari angka nol dan dilanjutkan ke angka berikutnya sampai akhir array.** Sebagai catatan bahwa indeks di dalam array dimulai dari **0 sampai dengan (ukuranArray-1).**

Sebagai contoh, pada array yang kita deklarasikan tadi, kita mempunyai:

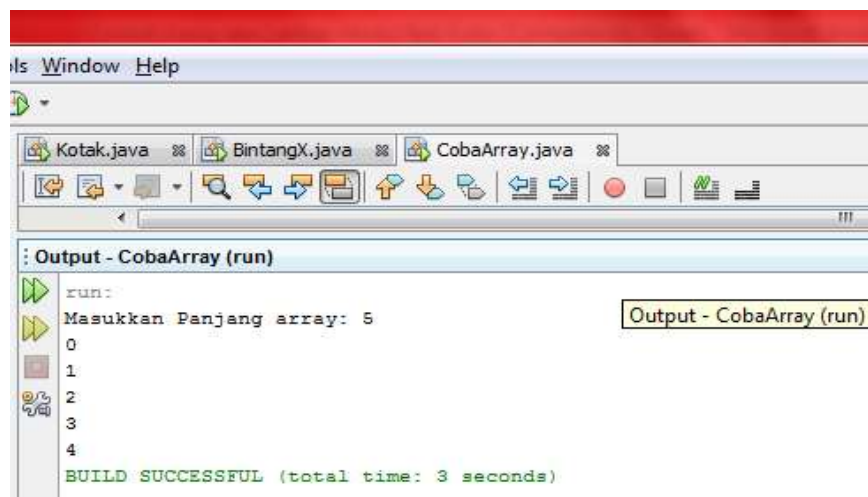
```
//memberikan nilai 10 kepada elemen pertama array
ages[0] = 10;
//mencetak elemen array yang terakhir
System.out.print(ages[99]);
```

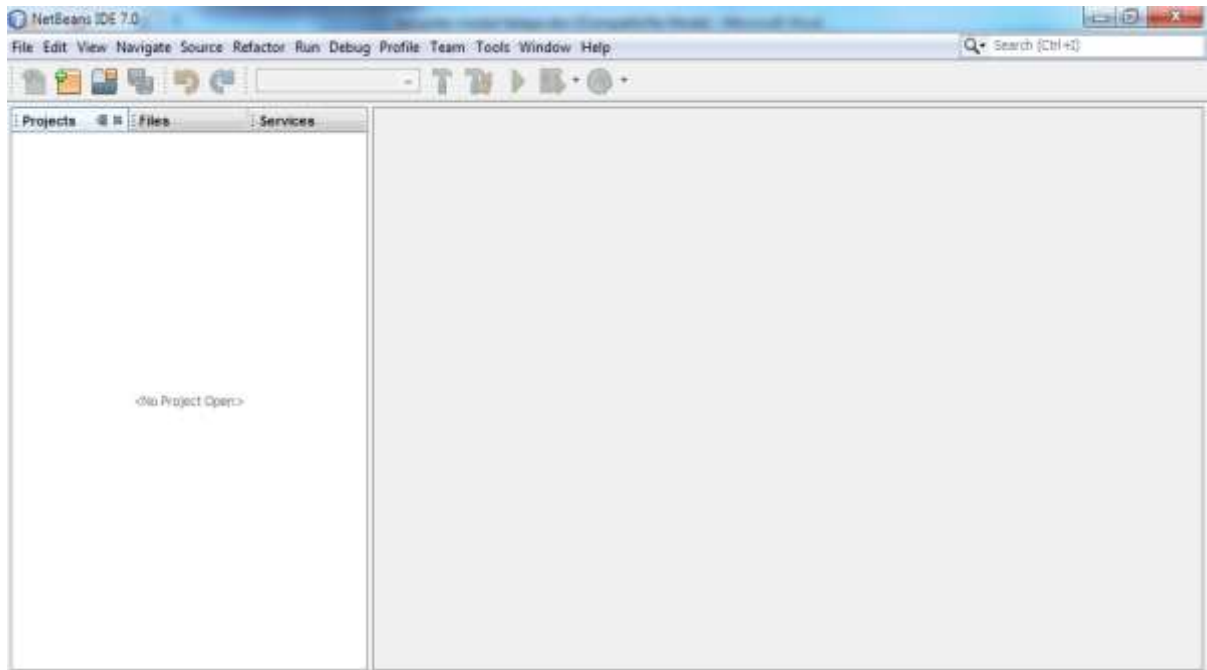
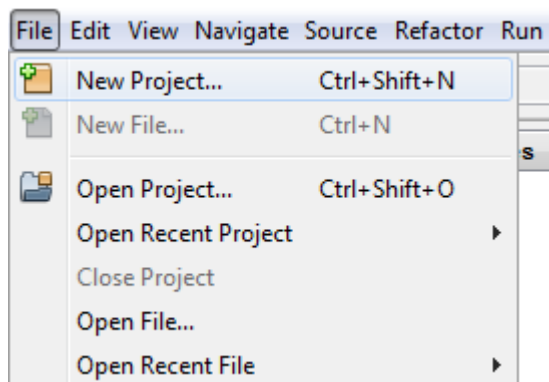
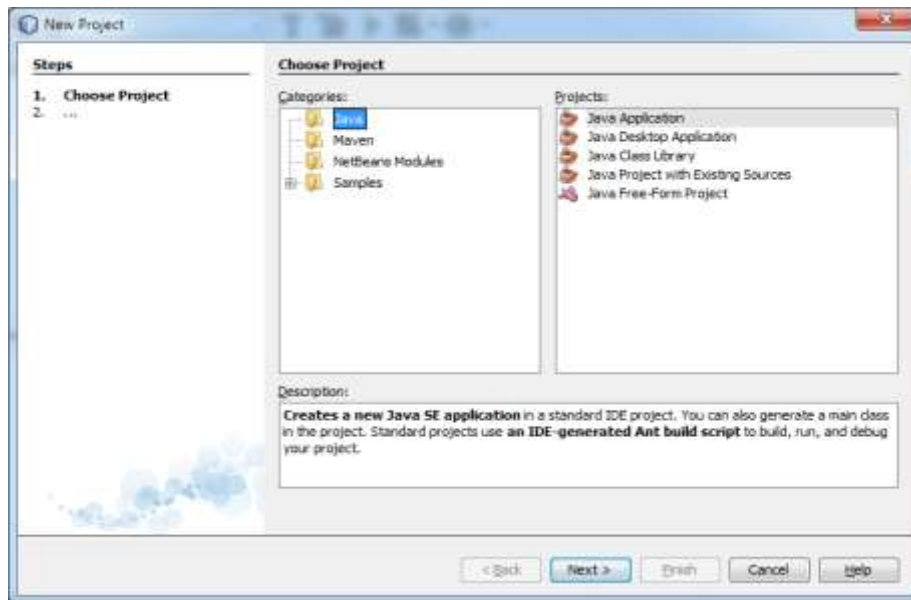
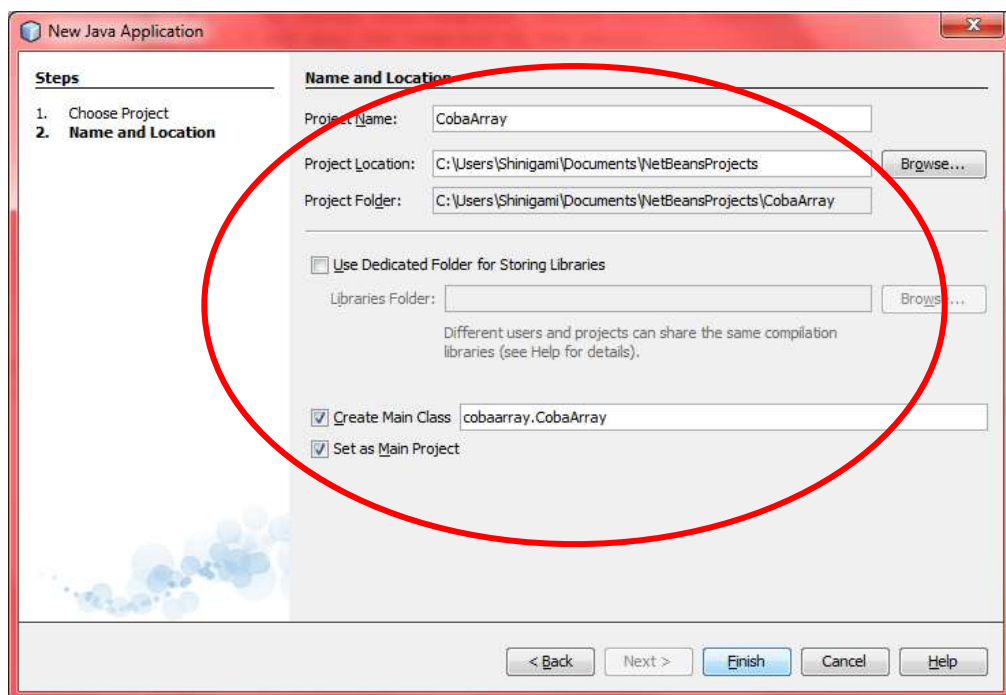
Perlu diperhatikan bahwa sekali array dideklarasikan dan dikonstruksi, nilai yang disimpan dalam setiap anggota array akan diinisialisasi sebagai nol. Oleh karena itu, apabila Anda menggunakan tipe data seperti String, array tidak akan diinisalisasi menjadi string kosong “”. Untuk itu Anda tetap harus membuat String array secara eksplisit.

Berikut ini adalah contoh kode untuk mencetak seluruh elemen di dalam array. Dalam contoh ini digunakanlah pernyataan *for loop*, sehingga kode kita menjadi lebih pendek.

```
public class ArraySample{
public static void main( String[] args ){
int[] ages = new int[100];
for( int i=0; i<100; i++ ){
System.out.print( ages[i] );
}
}
}
```

P4.2 Contoh Kasus

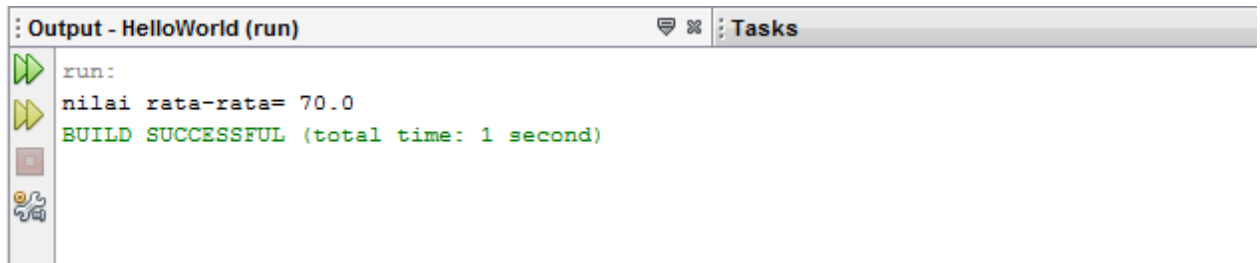
Buat program menggunakan array untuk menampilkan output:

Langkah-langkah Pengerjaan:

1. Klik icon Netbeans 7.0 pada desktop, hingga muncul tampilan seperti pada gambar di bawah ini

2. Klik menu file dan pilih New Project

3. Pada New Project Wizard, klik kategori Java dan pilih Java Application kemudian klik Next

4. Kemudian langkah berikutnya ialah :
   - Pada field **Project Name** ketikkan CobaArray
   - Pada field **project Location**, secara default akan disimpan pada direktori yang telah ditentukan.
   - Pada field **Create Main Class** ketikkan cobaarray.CobaArray (pada Netbeans nama class mengikuti nama project yang kita ketikkan)
   - Kemudian set checkbox pada **Set as main project dan Create Main Class**
   - Klik **Finish**

5. Selanjutnya ketikkan kode program di bawah ini pada code editor.

```
package cobaarray;

import java.io.*;
public class CobaArray {

    public static void main(String[] args) throws Exception{
        // TODO code application logic here
        String PanjangStr;
        int i, Panjang;

        BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

            System.out.print("Masukkan Panjang array: ");
            PanjangStr =br.readLine();
            Panjang = Integer.parseInt(PanjangStr);

            int[] bil = new int[Panjang];   // menyimpan nilai variabel panjang menjadi indeks dari
array bil

            /* perulangan for i melakukan perulangan dari 0 sampai sebanyak
             * indeks dari array bil */
            for( i=0; i<bil.length; i++ )
            {
                bil[i] = i; // menyimpan nilai perulangan i ke indeks array bil ke-i
                System.out.println( bil[i] );   // mencetak nilai dari indeks array bil ke-i
            }
    }
}
```

6. Build project tersebut dengan memilih menu Run – Build Main Project, atau dengan menggunakan hotkey F11

7. Jika tidak ada kesalahan (BUILD SUCCESSFUL), jalankan project tersebut dengan memilh menu Run – Run Main Project, atau dengan menggunakan hotkey F6

P4.3 Latihan

Berikut ini adalah program untuk mencari nilai rata-rata. Lengkapi kode program berikut dengan mengisi titik-titik yang berwarna merah.

```
package helloworld;

.......................................... //pendefinisian class array2 {
   public static void main(String[] args)
{
...............................//buatlah deklarasi array dengan variable X bertipe
data integer yang terdiri dari tiga bilangan

double rata=0.0;
..................................//buatlah pendeklarasian dengan perintah for
untuk melakukan perulangan dari setiap nilai array yang di deklarasikan
rata += x[i];
rata /=x.length;
System.out.println("nilai rata-rata= " + rata);
}
}
```

Output dari program di atas adalah:

```
Output - HelloWorld (run)        Tasks
run:
nilai rata-rata= 70.0
BUILD SUCCESSFUL (total time: 1 second)
```

Jawaban:

1. Jalankan Netbeans Anda
2. Lakukan langkah-langkah pengerjaan seperti contoh kasus sebelumnya.
3. Pada code editor Netbeans, ketikkan program berikut:

```
package helloworld;
public class array2 {
   public static void main(String[] args)
{
int[] x = {70,80,60};

double rata=0.0;
for(int i=0;i<x.length;i++)
rata += x[i];
rata /=x.length;
System.out.println("nilai rata-rata= " + rata);
}
}
```

P4.4 Daftar Pustaka

Naughton, Patrick, *Java Handbook: Konsep Dasar Pemrograman Java*, Andi, Yogyakarta, 1996.

Gary Cornell dan Cay S.Horstmann, *Core Java edisi Indonesia*, Andi Yogyakarta, 1997.

ANuff, Penuntun Pemrograman Java, Andi Yogyakarta, 1997.

Kadir, Abdul. Dasar Pemrograman Java 2. Andi. Yogyakarta, 2008.

Riyanto, Suprapto, *Pengembangan Aplikasi Manajemen Database dengan Java*, Penerbit Gava Media, Yogyakarta, 2008.

http://netbeans.org/kb/docs/java/quickstart.html